

REMARKS/ARGUMENTS

In the Final Office Action, the Examiner has maintained the rejection of claims under 35 U.S.C. 102(a) as being anticipated by the International Publication Number WO 02/091178 A2 (*Sharma et al.*) The Examiner's rejection is fully traversed below.

Initially, it is respectfully submitted that the Examiner has asserted that *Sharma et al.* "suggests," or "fairly suggests" a number of the claimed features which were submitted by the Applicant as patentable subject matter over *Sharma et al.* in the amendment dated June 3, 2005 (Final Office Action, page 3, I, III, and VI). The claims, however, stand rejected under 35 U.S.C. 102(e). Accordingly, it is respectfully submitted that the Examiner's rejection under 35 U.S.C. 102 (e) is clearly improper because the Examiner has merely asserted that these features are suggested or fairly suggested by *Sharma et al.*

Further, it is respectfully submitted that *Sharma et al.* does NOT constitute 103 art within the meaning of U.S.C. 103 because its subject matter and claimed invention were, at the time of invention, subject to an assignment to the same entity, namely, Sun Microsystems, Inc. (see 35 U.S.C. 103(c)). As such, *Sharma et al.* cannot be the basis of an obviousness rejection under 35 U.S.C. 103.

Still further, it is respectfully submitted that claim 1 is patentable over *Sharma et al.* for at least these reasons:

a) *Sharma et al.* does NOT teach loading an online upgrade module that includes a first container based software component, an online upgrade listener and online upgrade specification (claim 1)

In the Final Office Action, the Examiner asserts that *Sharma et al.* "fairly suggests" the loading feature recited in claim 1. To support this assertion, the Examiner has made reference to page 5, lines 1-17 (which is reproduced below) and Figure 3 of *Sharma et al.*

Sharma et al. in pertinent part states:

Broadly speaking, the present invention fills these needs by providing an upgrading mechanism that allows online upgrades of state schemas, including the managed state, both application specific and session related state, without disrupting the functionality of a running application during the upgrade process. In one embodiment, a

method for upgrading schema for a managed state for a Java based application is disclosed. A Java module is executed on a server, where the Java module includes at least one original entity bean and at least one original state object in communication with the original entity bean. The original state object (part of the representation of managed state schema) stores a state of the original entity bean. Then, an upgraded state object is generated and the state stored in the original state object is transferred to the upgraded state object. In this manner, state management for the original entity bean can be provided using the upgraded state object. In addition, an upgraded entity bean is made operational and the state of the upgraded entity bean is managed using the upgraded state object. Thereafter, both the original entity bean and the original state object can be disabled. In this manner, the functionality of the Java module is not disrupted when the upgrade of state schema is performed. Also, the functionality of the Java application is not disrupted when the Java module is upgraded. (*Sharma et al.*, page 5, lines 1-17)

Clearly, the upgrading mechanism described above does NOT teach or suggest the loading feature recited in claim 1. Accordingly, it is respectfully submitted that the Examiner's rejection is Improper for at least this reason.

b) *Sharma et al.* does NOT teach performing online upgrade operations in multiple states including an upgrade prepare stage, a pre-upgrade stage, one or more upgrade operations, a post-upgrade stage and a commit stage (claim 1)

In the Final Office Action, the Examiner has asserted that page 8, line 6-12 of *Sharma et al.* teaches this feature.

Sharma et al. in pertinent part states:

This subsystem is further capable of upgrading the managed state schema of an application using state objects, while the application continues to process requests. In the following description, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, to one skilled in the art that the present invention may be practiced without some or all of these specific details. In other instances, well known process steps have not been described in detail in order not to unnecessarily obscure the present invention. (*Sharma et al.* page 8, lines 6-12)

Clearly, "upgrading a managed state schema of an application using state objects" does NOT teach performing online upgrade operations in multiple stages. Therefore, it is respectfully submitted that the Examiner's rejection is improper for this additional reason.

c) *Sharma et al.* does NOT teach a pre-upgrade stage comprising: loading one or more listener classes associated with online upgrade listener, instantiating the one or more listeners associated with the online upgrade listener, and performing one or more callbacks via the online upgrade listener (Claim 1).

It is noted that *Sharma et al.* in pertinent part states:

The RSM of the embodiments of the present invention can generate concrete implementation classes for entity beans 304. The concrete classes generated by the RSM are responsible for managing the recoverable state of the entity beans 304. The RSM also provides implementation of collection classes that are used in managing container-managed relationships. By providing implementation of the getter and setter methods of the corresponding abstract classes, the RSM can implement the entity bean 304 classes. The RSM can also manage the mapping between primary keys and EJB objects, and can store the recoverable references to the remote and home interfaces of other EJBs. (*Sharma et al.*, page 10, lines 24-30)

During pre-deployment of an EJB module 302, the RSM maps the abstract schema of entity beans 304 classes to a physical schema used by the RSM. To perform this form of schema mapping, the RSM can use a deployment descriptor of EJB components. The RSM generates concrete implementations for the entity bean 304 classes defined as abstract classes by the application developer. A concrete implementation class includes the code that implements the setter and getter methods for container-managed fields and relationships based on the RSM mechanism. (*Sharma et al.*, page 11, lines 16-22)

However, it is respectfully submitted that the sections noted above which are the basis for the Examiner's rejection do NOT teach loading or instantiating listeners associated with an online upgrade listener, or performing call backs via the online upgrade listener. Accordingly, it is respectfully submitted that the Examiner's rejection is improper yet for these additional reasons.

d) *Sharma et al.* does not teach: (i) assuring successful draining of an older version of an application program, (ii) performing one or more callbacks, (iii) unloading the older version of the application program, and (iv) conveying information, about the assuring (i), performing (ii), or unloading(iii) features to a management entity (Claim 1).

In the Final Office Action, the Examiner has asserted that page 5, lines 7-17 and page 18, lines 13-20 of *Sharma et al.* "fairly suggests" these features. However, it is respectfully submitted that the cited sections of *Sharma et al.* which are the basis for the Examiner's rejection do NOT teach these features. Accordingly, it is respectfully submitted that the Examiner's rejection is improper still for the additional reasons.

e) The Examiner's rejection is improper for the following additional reasons:

The Examiner has NOT addressed several other claimed features. These features include: performing a callback to determine whether a first application program (which is an update to second program) is ready for service, performing a redirect callback to the first application program, and conveying information to a management entity about the loading or the callbacks (claim 20).

CONCLUSION

Based on the foregoing, it is submitted that all pending claims are patentably distinct over the cited art of record. Additional limitations recited in the independent claims or the dependent claims are not further discussed because the limitations discussed above are sufficient to distinguish the claimed invention from the cited art. Accordingly, Applicant believes that all pending claims are allowable and respectfully requests a Notice of Allowance for this application from the Examiner.

Applicants hereby petition for an extension of time which may be required to maintain the pendency of this case, and any required fee for such extension or any further fee required in connection with the filing of this Amendment is to be charged to Deposit Account No. 500388 (Order No. SUN1P849). Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP



R. Mahboubian
Reg. No. 44,890

P.O. Box 70250
Oakland, CA 94612-0250
(650) 961-8300